(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(51) 。 Int. Cl. [7]
H04L 9/30

(11) 공개번호 특2001-0090167
(43) 공개일자 2001년10월18일

(21) 출원번호 10-2000-0014822
(22) 출원일자 2000년03월23일

(71) 출원인 삼성전자 주식회사
윤종용
경기 수원시 팔달구 매탄3동 416

(72) 발명자 이경희
경기도수원시팔달구영통동벽적골한신아파트816동1205호

(74) 대리인 이영필
조혁근
이해영

심사청구 : 없음

(54) 패스워드를 기반으로 한 상호 인증 및 키 교환방법과 그장치

요약

본 발명은 패스워드를 기반으로 운영되는 시스템에 있어서, 해쉬 함수와 대칭 암호 시스템을 이용하여 클라이언트와 서버간의 상호 인증 서비스 및 키 교환 서비스를 제공할 수 있는 방법과 그 장치를 개시한다. 본 발명에 따른 방법은 클라이언트에서 발생된 랜덤값, 패스워드 검증자를 이용하여 얻어진 해쉬값와 랜덤값을 서버로 송출하면, 서버에서 클라이언트로부터 수신한 랜덤값과 서버에서 발생된 랜덤값을 이용하여 클라이언트와의 세션키를 생성하고, 생성된 세션키와 서버에서 발생한 패스워드 검증자 및 클라이언트로부터 수신한 랜덤값을 이용하여 해쉬값이 구해지면 구해진 해쉬값과 서버에서 발생된 랜덤값을 암호화하여 클라이언트로 송출하고, 이를 수신한 클라이언트는 수신한 정보와 자신이 발생한 랜덤값을 이용하여 서버와의 세션키를 생성하고, 생성된 세션키와 자신이 발생한 패스워드 검증자 및 랜덤값을 이용하여 다시 해쉬값을 구하고, 구해진 해쉬값과 서버로부터 수신한 해쉬값을 비교하여 서버가 인증되면 암호화 메시지를 송출하고, 서버는 수신한 암호화 메시지와 자신이 발생한 패스워드 검증자 및 클라이언트로부터 수신한 랜덤값을 이용하여 다시 해쉬값을 구하고, 구해진 해쉬값과 처음에 클라이언트로부터 수신한 해쉬값을 비교하여 클라이언트를 인증하는 단계로 이루어진다. 따라서 공격자로부터의 공격을 최대한 방어할 수 있고, 빠른 인증 및 키교환이 가능하다.

대표도
도 1

명세서

도면의 간단한 설명

도 1은 본 발명에 따른 패스워드를 기반으로 한 상호 인증 및 키 교환장치의 기능 블록도 이다.

도 2는 본 발명에 따른 패스워드를 기반으로 한 상호 인증 및 키 교환방법에 있어서 클라이언트 측의 동작 흐름도 이다.

도 3은 본 발명에 따른 패스워드를 기반으로 한 상호 인증 및 키 교환방법에 있어서 서버 측의 동작 흐름도 이다.

< 도면의 주요부분에 대한 부호의 설명>

100:클라이언트110:서버

101, 111:랜덤값 발생부102, 112:공개키 생성부

103:해쉬값 연산부104:패스워드 검증자 연산부

105, 114:송수신부106, 118:복호화부

107:세션키 생성 및 해쉬값 연산부108, 120:해쉬값 체크부

109:암호화 메시지 생성부113:세션키 생성부

115:패스워드 검증자 데이터 베이스116:제 1 해쉬값 연산부

117:암호화 정보 생성부119:제 2 해쉬값 연산부

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 컴퓨터 및 네트워크 환경에서 인증(authentication) 및 키(key)교환에 관한 것으로, 특히, 클라이언트(client)와 서버(server)간에 패스워드를 기반으로 한 상호 인증 및 키 교환방법과 그 장치에 관한 것이다.

컴퓨터 및 네트워크 환경에서 사용자에 대한 인증 시스템으로는 스마트 카드나 ID(IDentifier, 이하 ID라고 약함)카드와 같이 사용자가 갖고 있는 것을 이용하는 것과 홍체나 지문과 같이 사용자 그 자체를 인식하는 것 및 패스워드나 개인 ID(일명 PID(Personal ID))와 같이 사용자가 알고 있는 정보를 이용하는 것으로 분류할 수 있다.

그러나, 스마트 카드를 이용한 인증 시스템은 스마트 카드 자체의 보안 및 가격적인 문제 때문에 쉽게 사용되지 않고 있으며, 홍체 인식이나 지문 인식 등을 이용한 인증 시스템 역시 일부에서 사용되고 있긴 하나 가격이 워낙 비싸기 때문에 널리 사용되는데 어려움이 있는 반면에 패스워드 인증 시스템은 다른 인증 시스템에 비해 가격이 저렴하다는 이점으로 널리 사용되고 있다.

그러나, 패스워드는 사용자의 기억력을 기반으로 하고 있어 기억이 용이한 짧은 정보를 선택하는 경향이 있다. 이로 인하여 패스워드 인증 시스템은 다른 인증 시스템에 비해 해커(hacker)와 같은 공격자로부터 쉽게 공격당하는 단점이 있다.

즉, 패스워드를 데이터 베이스에 저장하여 이용하도록 구현한 패스워드 인증 시스템은 공격자가 시스템의 데이터 베이스에 접근만 하면 사용자의 패스워드를 쉽게 얻을 수 있으므로 공격자로부터의 공격(attack)에 거의 무방비 상태라 할 수 있다. 일방향 해쉬값(또는 해쉬 함수(hashing function))을 이용한 패스워드 인증 시스템의 경우는 사용자의 패스워드에 의한 해쉬값을 이용하므로 사용자의 패스워드를 알아내기가 어렵기는 하나 네트워크 환경에서 사용자가 입력한 패스워드를 서버로 전송할 때 클리어텍스트(cleartext) 형태로 전송하고 있어 통신상에서 사용자의 패스워드가 노출되어 공격자로부터 공격당하기 쉽다.

따라서 클리어텍스트 형태로 패스워드가 전송되는 것을 피하기 위해 체린지/응답(challenge and response)방식으로 운영되는 패스워드 인증 시스템이 제안되었다. 이 인증 시스템은 세션(session)을 설정할 때마다 서버가 랜덤한 값인 체린지를 송출하고, 클라이언트는 사용자의 패스워드와 수신한 체린지를 이용하여 가공한 응답을 서버로 송출하는 방식으로 인증 처리를 수행한다. 이로 인하여 공격자가 사전에 맞는 체린지-응답 쌍을 갖고 있을 지라도 다음 세션 설정시 사용되는 체린지가 달라 또 다른 응답을 필요로 하기 때문에 상술한 두 인증 시스템들에 비해 공격자로부터의 공격에 견딜 수 있다.

그러나, 인증 시스템의 개발 못지 않게 공격자의 공격방법도 날로 다양해짐에 따라, 상술한 체린지/응답방식으로 운영되는 인증 시스템은 상호 인증 및 세션 키를 생성할 수 있는 기능이 지원되지 않으며, 완벽한 순방향 비밀(perfect forward secrecy) 성질을 제공하지 못하는 단점으로 인하여 맨 인 더 미들(man-in-the-middle)방식과 같은 공격방법에 의해 공격당하였다. 맨 인 더 미들 방식은 서버와 클라이언트 사이에 송/수신되는 메시지를 공격자 자신의 메시지로 바꾸어 버리는 방식으로 공격하는 방법이다. 즉, 공격자의 메시지가 클라이언트에게는 서버의 메시지인 것처럼 여겨지며 서버에게는 클라이언트의 메시지인 것처럼 여겨지도록 공격하는 방법이다.

현재, 패스워드 기반의 인증 시스템에 대한 아주 강력한 공격방법으로 알려진 것은 패스워드 추측 공격(password guessing attacks)방식이다. 이 방식은 공격자가 클라이언트와 서버간의 통신을 저장하고 사전에 저장된 통신내용과 일치하는 패스워드를 찾아내는 오프라인(off-line) 공격방식으로, 클라이언트가 패스워드를 선택하는 자유도의 제약에 기초한 것이다. 즉, k비트 크기를 갖는 패스워드를 선택할 때, 각각의 k비트에 0과 1이 나올 확률이 0.5일 때 k비트 패스워드는 임시 랜덤 키와 같아지며 이를 추측하는 것은 2^k개의 랜덤한 패스워드 후보 리스트를 만드는 것과 같으며 이는 가능한 모든 가지 수를 시도하는 공격방식인 비라우트-포스 공격(broute-force attack)에 의한 것과 같은 것이다. 하지만 사용자가 패스워드를 선택할 때, 랜덤하게 선택하는 것은 거의 불가능하다. 이 때문에 오프라인 패스워드 추측 공격에 노출되는 실마리를 제공한다.

따라서, 이러한 오프라인 패스워드 추측 공격에 견딜 수 있는 패스워드 기반의 인증 프로토콜이 다양하게 제안되었다. 즉, 1993년 AT & T의 Bellovin이 EKE(Encrypted Key Exchange) 프로토콜을 제안하였는데, 이 EKE 프로토콜은 패스워드 추측 공격을 견딜 수 있는 강력한 인증 프로토콜이기는 하나 많은 메시지 라운드 및 미비한 기능들이 존재한다. 이에 Steiner가 1995년에 EKE에 PFS(Perfect Forward Secrecy) 기능을 추가하여 3-라운드(3-round)로 운영되는 인증 프로토콜을 제안하였고, 1997년에는 Stefan Lucks에 의해 EKE에 비해 공개키가 암호화되지 않고 그대로 전송되어지며 같은 공개키/비밀키 쌍이 많은 프로토콜 수행에 적합한 인증 시스템이 제안되었으며, RSA(Rirest Shamir Adleman)를 기반으로 한 변형 암호 알고리즘을 사용하는데 적합한 인증 시스템도 제안되었다.

또한, 공개키 암호 시스템과 패스워드를 이용한 프로토콜이 제안되었으나 이들은 인증 및 키 교환을 위해 공개키 연산을 많이 수행해야 한다. 1998년에 Shai Halevi와 Hugo Krawczyk에 의해 이와 비슷한 공개키 암호화 및 패스워드 프로토콜이 제안되었고, 1998년에는 Thomas Wu.에 의해 안정된 원격 패스워드 프로토콜이 제안되었으나 이는 비밀 공개키 연산측면에서 효율적이긴 하지만 4-라운드(round) 메시지 형태로 구성되어 있다.

최근에는 패스워드에 기반한 인증 프로토콜을 이용해서 네트워크에서 비밀키를 다운 로드하는 프로토콜이 제안되었으나 이 프로토콜은 새로운 프로토콜을 제안하는 것이 아니고 기존의 프로토콜을 이용해서 비밀키를 다운 로드하는 방법을 제안하고 있다. 가변길이의 패스워드를 이용할 수 있는 메카니즘도 제안되었으나 네트워크상에서 인증하는 메카니즘이 아니며, 한 호스트내에서 패스워드에 기반한 사용자 인증에 초점을 맞추고 있다. 또한, P. Mackenzie와 R. Swaminathan에 의해 제안된 패스워드 ID에 의해 안전한 망 인증방식이 제안되기도 하였는데 이 방식은 입증 가능한 보안(provable security) 성질을 갖도록 설계된 것이 특징이다.

발명이 이루고자 하는 기술적 과제

본 발명은 상술한 바와 같이 패스워드 추측 공격에 견딜 수 있는 다양한 인증 프로토콜이 제안되고 있는 추세에 따라 안출된 것으로, 패스워드를 기반으로 운영되는 시스템에 있어서, 해쉬 함수와 대칭 암호 시스템을 이용하여 클라이언트와 서버간의 상호 인증 서비스 및 키 교환 서비스를 제공할 수 있는 패스워드를 기반으로 한 상호 인증 및 키 교환방법과 그 장치를 제공하는데 그 목적이 있다.

본 발명의 다른 목적은 사용자의 패스워드에 대한 익명성(anonymity)을 제공하고 인증 및 키 교환 서비스를 제공하는 시간을 단축시킬 수 있는 패스워드를 기반으로 한 상호 인증 및 키 교환방법과 그 장치를 제공하는데 있다.

발명의 구성 및 작용

본 발명이 이루고자 하는 기술적인 과제를 해결하기 위한 방법은, 클라이언트와 서버간 통신 환경에 있어서, 클라이언트에서 인증을 위해 발생한 랜덤값과 패스워드 검증자를 이용하여 제 1 해쉬값을 연산하여 서버로 송출하는 제 1 단계; 서버는 클라이언트로부터 수신한 랜덤값과 서버에서 발생된 랜덤값을 이용하여 클라이언트와 서버간의 세션키를 생성하는 제 2 단계; 서버는 세션키와 서버에서 발생된 패스워드 검증자 및 클라이언트로부터 수신한 랜덤값을 이용하여 제 2 해쉬값을 연산하는 제 3 단계; 제 2 해쉬값과 서버에서 발생된 랜덤값을 암호화하여 클라이언트로 송출하는 제 4 단계; 클라이언트는 서버로부터 암호화한 정보가 수신되면, 클라이언트에서 발생한 랜덤값과 서버로부터 수신한 암호화 정보를 이용하여 클라이언트와 서버간의 세션키를 생성하는 제 5 단계; 클라이언트는 클라이언트에서 발생한 랜덤값과 패스워드 검증자 및 제 5 단계에서 생성된 세션키를 이용하여 제 3 해쉬값을 연산하는 제 6 단계; 제 3 해쉬값과 서버로부터 수신한 암호화 정보에 실려 있는 제 2 해쉬값을 비교하여 서버를 인증하는 제 7 단계; 제 7 단계 수행결과, 클라이언트로부터 암호화 메시지가 수신되면, 서버는 수신된 암호화 메시지와 서버에서 발생된 패스워드 검증자 및 클라이언트로부터 수신한 랜덤값을 이용하여 제 4 해쉬값을 연산하는 제 8 단계; 제 4 해쉬값과 제 1 해쉬값을 비교하여 클라이언트를 인증하는 단계를 포함하는 포함하는 것이 바람직하다.

본 발명이 이루고자 하는 기술적인 과제를 해결하기 위한 장치는, 클라이언트와 서버간 통신 환경에 있어서, 클라이언트는 인증을 위해 클라이언트에서 발생한 랜덤값과 패스워드 검증자를 해쉬 연산하여 얻어진 제 1 해쉬값과 랜덤값을 상기 서버로 송출하고, 서버로부터 암호화 정보가 수신되면 서버와의 세션키를 생성하고, 세션키와 랜덤값 및 패스워드 검증자를 해쉬 연산하여 얻어진 제 2 해쉬값과 제 1 해쉬값을 비교하여 서버를 인증하도록 구성되고; 서버는 인증을 위해 서버에서 발생된 랜덤값과 클라이언트에서 수신된 랜덤값을 이용하여 클라이언트와의 세션키를 생성하고, 생성된 세션키와 서버에서 발생된 패스워드 검증자 및 클라이언트에서 수신한 랜덤값을 해쉬 연산하여 얻어진 제 3 해쉬값과 서버에서 발생된 랜덤값을 암호화하여 클라이언트로 송출하고, 클라이언트로부터 암호화 메시자가 수신되면 제 4 해쉬값을 구하여 클라이언트로부터 수신한 제 1 해쉬값과 비교하여 클라이언트를 인증하도록 구성되는 것이 바람직하다.

이하, 첨부된 도면을 참조하여 본 발명을 상세히 설명한다.

도 1은 사용자의 패스워드를 기반으로 운영되는 시스템에 있어서 본 발명에 따른 패스워드를 기반으로 한 상호 인증 및 키 교환장치의 기능 블록도로서, 네트워크를 통해 상호 통신이 가능한 클라이언트(100) 및 서버(110)로 구성된다.

클라이언트(100)는 사용자측에서 운영되는 시스템으로서, 랜덤값 발생부(101), 공개키 생성부(102), 해쉬값 연산부(103), 패스워드 검증자 연산부(104), 송수신부(105), 복호화부(106), 세션키 생성 및 해쉬값 연산부(107), 해쉬값 체크부(108) 및 암호화 메시지 생성부(109)로 구성된다. 서버(110)는 랜덤값 발생부(111), 공개키 생성부(112), 세션키 생성부(113), 송수신부(114), 패스워드 검증자 데이터 베이스(115), 제 1 해쉬값 연산부(116), 암호화 정보 생성부(117), 복호화부(118), 제 2 해쉬값 연산부(119) 및 해쉬값 체크부(120)로 구성된다.

도 1에 도시된 클라이언트(100)의 랜덤값 발생부(101)는 사용자로부터 인증 요구신호가 인가되면, 저장되어 있는 다수의 값들에서 클라이언트/사용자의 임시 개인 키(private key) x_A, 클라이언트/사용자의 컨파운더(confounder) c_A 및 랜덤값 r 에 해당되는 정보를 각각 랜덤하게 골라서 발생하도록 구성된다. 랜덤값 발생부(101)에서 발생된 개인 키 x_A는 공개키 생성부(102) 및 세션키 생성 및 해쉬값 연산부(107)로 전송되고, 컨파운더 c_A는 해쉬값 연산부(103)로 전송되고, 랜덤 값 r 은 해쉬값 연산부(103), 송수신부(105) 및 세션키 생성 및 해쉬값 연산부(107)로 각각 전송된다.

공개키 생성부(102)는 랜덤값 발생부(101)로부터 전송된 개인 키 x_A에 대응되는 공개 키(public key) y_A를 생성한다. 공개 키 y_A는 수학식 1과 같은 모듈러 멱승 연산에 의해 생성된다.

수학식 1

$$y_A = g^{x_A}$$

수학식 1에서 g 는 mod(modulur) n에 대한 원시원이다. 생성된 공개 키 y_A는 해쉬값 연산부(103) 및 송수신부(105)로 각각 전송된다.

패스워드 검증자 연산부(104)는 사용자로부터 패스워드 정보가 인가되면, 수학식 2와 같은 모듈러 멱승 연산에 의해 얻어진 값을 대응되는 패스워드 검증자(verifier) v 로서 출력한다. 수학식 2에서 $H(P)$는 사용자가 입력한 패스워드 정보이다.

수학식 2

$$v = g^{H(P)}$$

해쉬값 연산부(103)는 랜덤값 발생부(101), 공개키 생성부(102) 및 패스워드 검증자 연산부(104)로부터 각각 전송된 값들을 연산하여 해쉬값(또는 해쉬함수) $h(y_A, r, c_A)$을 구한다. 해쉬값 $h(y_A, r, c_A)$을 구하는 방식은 기존에 알려진 방식과 동일하다. 여기서 구해진 해쉬값 $h(y_A, r, c_A)$을 z_1이라 한다. 구해진 해쉬값 z_1은 송수신부(105)로 전송된다.

송수신부(105)는 서버(110)로 세션 요구신호(session request)를 송출할 때, 해쉬값 연산부(103)로부터 전송된 해쉬값 z_1과 랜덤값 발생부(101)로부터 전송된 랜덤값 r 및 공개키 생성부(102)로부터 전송된 공개키 y_A을 각각 서버(110)로 송출한다.

복호화부(106)는 송수신부(105)를 통해 서버(110)로부터 암호화 정보 z_2가 수신되면, 이를 복호화한 뒤, 서버(110)에서 생성한 공개 키y_B에 대한 복호화 정보는 세션 키 생성부 및 해쉬값 연산부(107)로 전송하고, 제 1 해쉬값 h_1에 대한 복호화 정보는 해쉬값 체크부(108)로 전송한다. 암호화 정보 z_2는 서버(110)에서 생성한y_B와 제 1 해쉬값 h_1을 암호화한 정보이다.

세션 키 생성 및 해쉬값 연산부(107)는 복호화부(106)로부터 전송된 복호화된 공개키 $y_B(=g^{x_B})$와 랜덤값 발생부(101)로부터 제공되는 개인 키x_A을 수학식 3과 같이 연산하여 세션 키(K)를 생성한다.

수학식 3

$$K=g^{x_A x_B}$$

그리고, 생성된 세션 키K와 패스워드 검증자 연산부(104)로부터 전송된 패스워드 검증자 및 랜덤값 발생부(101)로부터 제공되는 랜덤값 를 이용하여 해쉬값h_1 $(=h(r,v,K))$를 구한다. 구해진 해쉬값h_1은 해쉬값 체크부(108)로 전송한다.

해쉬값 체크부(108)는 복호화부(106)에서 제공되는 서버(110)에서 연산된 해쉬값(h_1)과 세션 키 및 해쉬값 연산부(107)로부터 전송되는 해쉬값h_1이 동일한 지를 비교하여 세션 키에 대한 인증작업을 수행한다. 비교결과, 두 해쉬값이 동일하면($h_1=h_1$), 해쉬값 체크부(108)는 인증작업 완료 메시지를 출력하고 두 해쉬값이 동일하지 않으면($h_1 \neq h_1$), 서버(110)와의 연결(connection)이 실패(fail)처리되도록 송수신부(105)로 이를 통보한다.

암호화 메시지 생성부(109)는 해쉬값 체크부(108)로부터 암호화 메시지 생성을 요구하는 신호가 전송되면, 랜덤값 발생부(101)로부터 제공된 컨파운더c_A와 세션 키K를 이용하여 암호화 메시지$M=E_{y_B}(c_A,K)$를 생성한다. 생성된 암호화 메시지(M)는 송수신부(105)로 전송된다.

송수신부(105)는 서버(110)와 클라이언트(100)간에 인증 및 키 교환을 위한 정보를 송수신할 수 있도록 구성된다.

한편, 서버(110)에 구비되어 있는 송수신부(114)는 서버(110)와 클라이언트(100)간에 인증 및 키 교환을 위한 정보를 송수신할 수 있도록 구성된다.

랜덤값 발생부(111)는 서버(110)가 수용하고 있는 클라이언트(100)로부터의 세션 요구를 미리 예측한 시점에서 저장되어 있는 다수의 랜덤값에서 서버(110)의 임시 개인키 정보에 해당되는x_B을 발생한다. 발생된 개인키 정보x_B는 공개키 생성부(112)로 전송된다.

공개키 생성부(112)는 사용자 단말기(100)에 구비되어 있는 공개키 생성부(102)와 같은 방식으로 인가된 개인키 정보를 이용한 모듈러 멱승연산(g^{x_B})결과를 공개키y_B로서 출력한다. 출력된 공개키y_B는 세션키 생성부(113)로 전송된다.

세션키 생성부(113)는 송수신부(114)로부터 전송된 클라이언트(100)에서 생성된 사용자의 임시 개인키 정보 x_A와 공개키 생성부(112)로부터 전송된y_B를 상술한 수학식 3과 같이 연산하여 세션키K를 생성한다. 생성된 세션키K는 제 1 해쉬값 연산부(116)로 전송된다.

제 1 해쉬값 연산부(116)는 인가된 세션키와 클라이언트(100)에 해당되는 패스워드 검증자 v 및 클라이언트(100)로부터 수신한 랜덤값 r를 이용한 해쉬 함수$h(r,v,K)$에 의한 해쉬값h_1를 생성한다. 이 때, 패스워드 검증자 v는 패스워드 검증자 데이터 베이스(115)로부터 제공된다. 패스워드 검증자 데이터 베이스(115)는 서버(110)가 수용하는 모든 클라이언트의 패스워드 검증자 정보를 사전에 저장하고 있다가 송수신부(114)를 통해 수신된 공개키y_A가 인가되면, 대응되는 패스워드 검증자를 출력하는 것으로, 패스워드 검증자 제공부 역할을 한다.

암호화 정보 생성부(117)는 공개키 생성부(112)로부터 제공되는 공개키y_B와 제 1 해쉬값 연산부(116)에서 제공되는 해쉬값h_1을 이용하여 암호화 정보z_2 ($=E_{y_B}(h_1)$)를 생성한다. 생성된 암호화 정보는 송수신부(114)를 통해 해당되는 클라이언트(100)로 송출된다.

복호화부(118)는 송수신부(114)를 통해 수신된 암호화 메시지M을 복호화하여 암호화 메시지에 포함되어 있는 컨파운더c_A를 제 2 해쉬값 연산부(119)로 제공한다.

제 2 해쉬값 연산부(119)는 복호화부(118)에서 제공되는 컨파운더 c_A와 패스워드 검증자 데이터 베이스(115)에서 제공되는 패스워드 검증자v 와 송수신부(114)로부터 제공되는 공개키y_A을 이용하여 해쉬값z_1' ($=h(y_A,v,c_A)$)을 얻는다. 얻어진 해쉬값z_1'은 해쉬값 체크부(120)로 전송된다.

해쉬값 체크부(120)는 제 2 해쉬값 연산부(119)로부터 제공된 해쉬값 z_1'와 세션 요구시 송수신부(114)에서 수신하였던 해쉬값z_1이 동일한 지를 체크한다. 체크결과, 동일하면($z_1'=z_1$) 인증작업 완료 통보신호를 출력한다. 반면에 동일하지 않으면($z_1'\neq z_1$) 사용자 단말기(100)와의 연결이 실패처리되도록 송수신부(114)로 이를 통보한다.

상술한 도 1은 서버(110)의 개인키x_B및 공개키y_B을 생성하는 시점이 수용하고 있는 클라이언트(100)가 세션을 요구할 시점을 미리 예측하여 세션요구신호가 클라이언트(100)에서 발생되기 전에 생성되도록 구현된 경우를 예시하였으나 클라이언트(100)로부터 세션 요구신호를 수신한 후, 서버(100)의 개인키 x_B및 공개키y_B을 생성하도록 구현할 수도 있다.

도 2는 본 발명에 따른 패스워드를 기반으로 한 상호 인증 및 키 교환방법의 동작 흐름도중 클라이언트(100)의 동작 흐름도이고, 도 3은 서버(110)의 동작 흐름도 이다. 도 2 및 도 3을 참조하여 본 발명에 따른 방법을 상세하게 설명하면 다음과 같다.

먼저, 클라이언트(100)는 단계 201에서 사용자로부터 서버(110)와의 인증이 요청되면, 단계 203으로 진행되어 저장되어 있는 다수의 랜덤값들중 개인키 정보x_A, 컨파운더 정보c_A및 랜덤값 r을 랜덤하게 골라서 각각 발생하고, 발생된 개인키 정보x_A을 이용하여 해당되는 공개키 정보y_A을 상술한 수학식 1에 의한 연산으로 생성한다.

그 다음 단계 205에서 사용자의 패스워드가 입력되면, 단계 207에서 수학식 2와 같이 연산하여 패스워드 검증자 v 을 구한다. 그리고, 단계 209에서 공개키 정보 y_A, 패스워드 검증자 v 및 컨파운더 정보 c_A을 이용하여 해쉬값 z_1 (= $h(y_A,v,c_A)$)을 연산하고, 단계 211로 진행된다.

단계 211에서 클라이언트(100)는 해쉬값 z_1, 공개키 y_A 및 랜덤값 r 을 각각 서버(110)측으로 송출한다.

서버(110)로 상술한 값들을 송출한 후, 서버(110)로부터 암호화 정보(z_2)가 수신되면, 단계 213에서 단계 215로 진행되어 수신된 암호화 정보(z_2)를 복호화한다. 그리고, 단계 217에서 복호화된 서버(110)측의 공개키 정보 y_B을 이용하여 세션키 K 를 생성한다. 그리고 생성된 세션키 K, 클라이언트(100)에서 발생된 랜덤값 r, 패스워드 검증자 v 을 이용하여 해쉬값 h_1 (=$h(r,v,K)$)를 구한다.

그 다음, 단계 219에서 구해진 해쉬값 h_1 과 단계 215에서 복호화된 서버(110)로부터 수신된 해쉬값 h_1이 동일한 지를 비교한다. 비교결과, 동일하면 단계 221로 진행되어 암호화 메시지 $M=E_{y_B}(c_A,K)$ 을 생성하여 서버(110)측으로 송출한 뒤, 인증 및 키 교환작업을 종료한다. 그러나, 단계 219에서 비교한 결과, 생성한 세션 키와 수신한 세션 키가 동일하지 않으면 단계 223으로 진행되어 해당되는 연결에 대한 실패처리를 하고 인증 및 키 교환작업을 종료한다.

상술한 클라이언트(100)의 동작중 단계 201 및 단계 203과 단계 205 및 단계 207의 동작은 병렬적으로 수행되거나 단계 203과 단계 205에 의한 동작이 단계 201 및 단계 203에 의한 동작보다 선행되도록 구현될 수도 있다.

한편, 서버(110)는 단계 301에서 저장되어 있는 랜덤값중 서버(110)의 임시 개인키 정보 x_B에 해당되는 값을 랜덤하게 골라서 발생한다. 이 개인키 정보 x_B에 대한 발생시점은 서버(110)가 수용하고 있는 클라이언트(100)의 세션 요구를 미리 예측한 시점이 된다. 그리고, 발생된 임시 개인키 정보 x_B을 이용하여 공개키 y_B를 생성한다. 공개키 y_B를 생성하는 방식은 상술한 수학식 1과 같이 이루어진다.

단계 303에서 서버(110)는 임의의 클라이언트(100)로부터 해쉬값 z_1, 공개키 y_A 및 랜덤값 r 이 각각 수신되면, 단계 305로 진행되어 서버(110)에서 생성한 공개키 y_B와 수신한 클라이언트(100)의 공개키 y_A을 이용하여 세션키 K (= $g^{x_A x_B}$)를 상술한 수학식 3과 같이 생성한다.

단계 307에서 서버(110)는 제 1 해쉬값 h_1 (=$h(r,v,K)$)을 연산하고, 단계 309에서 제 1 해쉬값 h_1과 공개키 y_B을 암호화한 정보 z_1을 생성하고, 해당되는 클라이언트(100)로 송출한다.

그 후에 해당되는 클라이언트(100)로부터 대응되는 암호화 메시지($M=E_{y_B}(c_A,K)$)가 수신되면, 단계 311에서 단계 313으로 진행되어 수신된 암호화 메시지 M을 복호화하고, 단계 315에서 복호화된 정보를 이용하여 제 2 해쉬값 z_1'(= $h(y_A,v,c_A)$)을 연산한다.

그리고, 단계 317에서 단계 315에서 연산한 제 2 해쉬값 z_1'와 단계 303에서 수신한 해쉬값z_1이 동일한 지를 비교한다. 비교결과, 동일하면 단계 319로 진행되어 해당되는 클라이언트(100)와의 상호 인증 작업완료를 통보하고, 인증 및 키교환작업을 종료한다. 그러나, 단계 317에서 비교한 결과, 제 2 해쉬값 z_1'과 수신한 해쉬값z_1이 동일하지 않으면, 단계 321로 진행되어 해당되는 클라이언트(100)와의 연결실패처리를 한 뒤, 해당되는 인증 및 키교환 작업을 종료한다.

도 3에서의 서버(110)의 동작도 클라이언트(100)로부터 세션 요구가 수신되기 전에 랜덤값 및 공개키가 생성되는 방법으로 운영되나 이는 도 1에서 상술한 바와 같이 클라이언트(100)로부터 세션 요구신호가 수신된 후에 랜덤값 및 공개키가 생성되도록 구현될 수도 있다.

발명의 효과

상술한 바와 같이 본 발명에 따른 패스워드를 기반으로 한 상호 인증 및 키 교환방법과 그 장치는 인터넷 등 안전하지 않은 네트워크상에서 클라이언트와 서버간의 상호 인증이 가능하도록 구현하였을 뿐만 아니라 상호 인증과 키생성을 결합시켜 운영하고, 완벽한 순방향 안정성을 제공하며, 패스워드와 관련된 메시지가 네트워크상에서 송수신되지 않도록 구현함으로써, 맨 인 더 미들 공격방식과 같은 공격에 강력하게 방어할 수 있고 기존이 인증 알고리즘에 비해 인증 결과에 대한 신뢰도를 향상시키는 효과가 있다.

또한, 패스워드만을 이용하여 상대방을 인증하도록 구현하여 클라이언트는 서버와 관련된 어떠한 정보도 필요로 하지 않기 때문에 기존의 프로토콜에 비해 구현이 용이하고, 3-라운드 메시지 전달만으로 인증 및 키생성이 가능하도록 구현하여 메시지 전달을 최소화하였을 뿐 아니라 클라이언트 및 서버가 각각 한번의 모듈러 멱승연산을 수행하는 것을 제외하고 대부분 해쉬값 및 대칭(symmetric) 암호 알고리즘을 수행하도록 구현함으로써, 빠른 인증 및 키교환 기능을 제공하는 효과도 있다.

(57) 청구의 범위

청구항 1.

클라이언트와 서버간 통신 환경에 있어서,

상기 클라이언트에서 인증을 위해 발생한 랜덤값과 패스워드 검증자를 이용하여 제 1 해쉬값을 연산하여 상기 서버로 송출하는 제 1 단계;

상기 서버는 상기 클라이언트로부터 수신한 상기 랜덤값과 상기 서버에서 발생된 랜덤값을 이용하여 상기 클라이언트와 상기 서버간의 세션키를 생성하는 제 2 단계;

상기 서버는 상기 세션키와 상기 서버에서 발생된 패스워드 검증자 및 상기 클라이언트로부터 수신한 랜덤값을 이용하여 제 2 해쉬값을 연산하는 제 3 단계;

상기 제 2 해쉬값과 상기 서버에서 발생된 랜덤값을 암호화하여 상기 클라이언트로 송출하는 제 4 단계;

상기 클라이언트는 상기 서버로부터 암호화한 정보가 수신되면, 상기 클라이언트에서 발생한 랜덤값과 상기 서버로부터 수신한 암호화 정보를 이용하여 상기 클라이언트와 상기 서버간의 세션키를 생성하는 제 5 단계;

상기 클라이언트는 상기 클라이언트에서 발생한 랜덤값과 패스워드 검증자 및 상기 제 5 단계에서 생성된 세션키를 이용하여 제 3 해쉬값을 연산하는 제 6 단계;

상기 제 3 해쉬값과 상기 서버로부터 수신한 암호화 정보에 실려 있는 상기 제 2 해쉬값을 비교하여 상기 서버를 인증하는 제 7 단계;

상기 제 7 단계 수행결과, 상기 클라이언트로부터 암호화 메시지가 수신되면, 상기 서버는 수신된 상기 암호화 메시지와 상기 서버에서 발생된 패스워드 검증자 및 상기 클라이언트로부터 수신한 랜덤값을 이용하여 제 4 해쉬값을 연산하는 제 8 단계;

상기 제 4 해쉬값과 상기 제 1 해쉬값을 비교하여 상기 클라이언트를 인증하는 단계를 포함하는 패스워드를 기반으로 한 상호 인증 및 키 교환방법.

청구항 2.

제1항에 있어서, 상기 제 7 단계는 상기 제 3 해쉬값과 상기 제 2 해쉬값이 동일하면 상기 암호화 메시지를 송출한 뒤, 상기 서버에 대한 상기 클라이언트의 인증작업을 종료하고, 상기 제 8 단계는 상기 제 4 해쉬값과 상기 제 1 해쉬값이 동일하면 상기 클라이언트에 대한 상기 서버의 인증작업을 종료하는 단계로 이루어지는 패스워드를 기반으로 한 상호 인증 및 키 교환방법.

청구항 3.

제1항 또는 제2항에 있어서, 상기 클라이언트에서 발생되는 랜덤값에는 상기 클라이언트의 임시 개인키, 컨파운더, 랜덤값 및 상기 임시 개인키에 의해 발생되는 임시 공개키에 대한 정보가 포함되고, 상기 서버에서 발생되는 랜덤값에는 상기 서버의 임시 개인키 및 상기 임시 개인키에 의해 발생되는 임시 공개키에 대한 정보가 포함되는 패스워드를 기반으로 한 상호 인증 및 키 교환방법.

청구항 4.

제3항에 있어서, 상기 제 5 단계는 상기 암호화한 정보가 수신되면 복호화하여 상기 세션키 생성에 이용되도록 제공하는 단계를 더 포함하고, 제 8 단계는 상기 암호화 메시지가 수신되면 복호화하여 상기 제 4 해쉬값 연산에 이용되도록 제공하는 단계를 더 포함하는 패스워드를 기반으로 한 상호 인증 및 키 교환방법.

청구항 5.

제3항에 있어서, 상기 제 1 해쉬값은 상기 클라이언트의 임시 공개키, 상기 클라이언트에서 발생된 패스워드 검증자 및 컨파운더를 해쉬 연산하여 구하고, 상기 제 2 해쉬값은 상기 클라이언트로부터 수신한 랜덤값, 상기 서버에서 발생된 패스워드 검증자 및 상기 제 2 단계에서 생성된 세션키를 해쉬 연산하여 구하고, 상기 제 3 해쉬값은 상기 제 5 단계에서 생성된 세션키, 상기 클라이언트에서 발생된 패스워드 검증자 및 상기 클라이언트에서 발생된 랜덤값을 해쉬 연산하여 구하고, 제 3 해쉬값은 상기 암호화 메시지에 포함되어 있는 컨파운더 상기 서버에서 발생된 패스워드 검증자 및 상기 제 1 단계에서 수신한 상기 클라이언트의 임시 공개키를 해쉬 연산하여 구하는 것을 특징으로 하는 패스워드를 기반으로 한 상호 인증 및 키 교환방법.

청구항 6.

클라이언트와 서버간 통신 환경에 있어서,

상기 클라이언트는 인증을 위해 상기 클라이언트에서 발생한 랜덤값과 패스워드 검증자를 해쉬 연산하여 얻어진 제 1 해쉬값과 상기 랜덤값을 상기 서버로 송출하고, 상기 서버로부터 암호화 정보가 수신되면 상기 서버와의 세션키를 생성하고, 상기 세션키와 상기 랜덤값 및 패스워드 검증자를 해쉬 연산하여 얻어진 제 2 해쉬값과 상기 제 1 해쉬값을 비교하여 상기 서버를 인증하도록 구성되고;

상기 서버는 인증을 위해 상기 서버에서 발생된 랜덤값과 상기 클라이언트에서 수신된 랜덤값을 이용하여 상기 클라이언트와의 세션키를 생성하고, 생성된 세션키와 상기 서버에서 발생된 패스워드 검증자 및 상기 클라이언트에서 수신한 랜덤값을 해쉬 연산하여 얻어진 제 3 해쉬값과 상기 서버에서 발생된 랜덤값을 암호화하여 상기 클라이언트로 송출하고, 상기 클라이언트로부터 암호화 메시지가 수신되면 제 4 해쉬값을 구하여 상기 클라이언트로부터 수신한 제 1 해쉬값과 비교하여 상기 클라이언트를 인증하도록 구성되는 것을 특징으로 하는 패스워드를 기반으로 한 상호 인증 및 키 교환장치.

청구항 7.

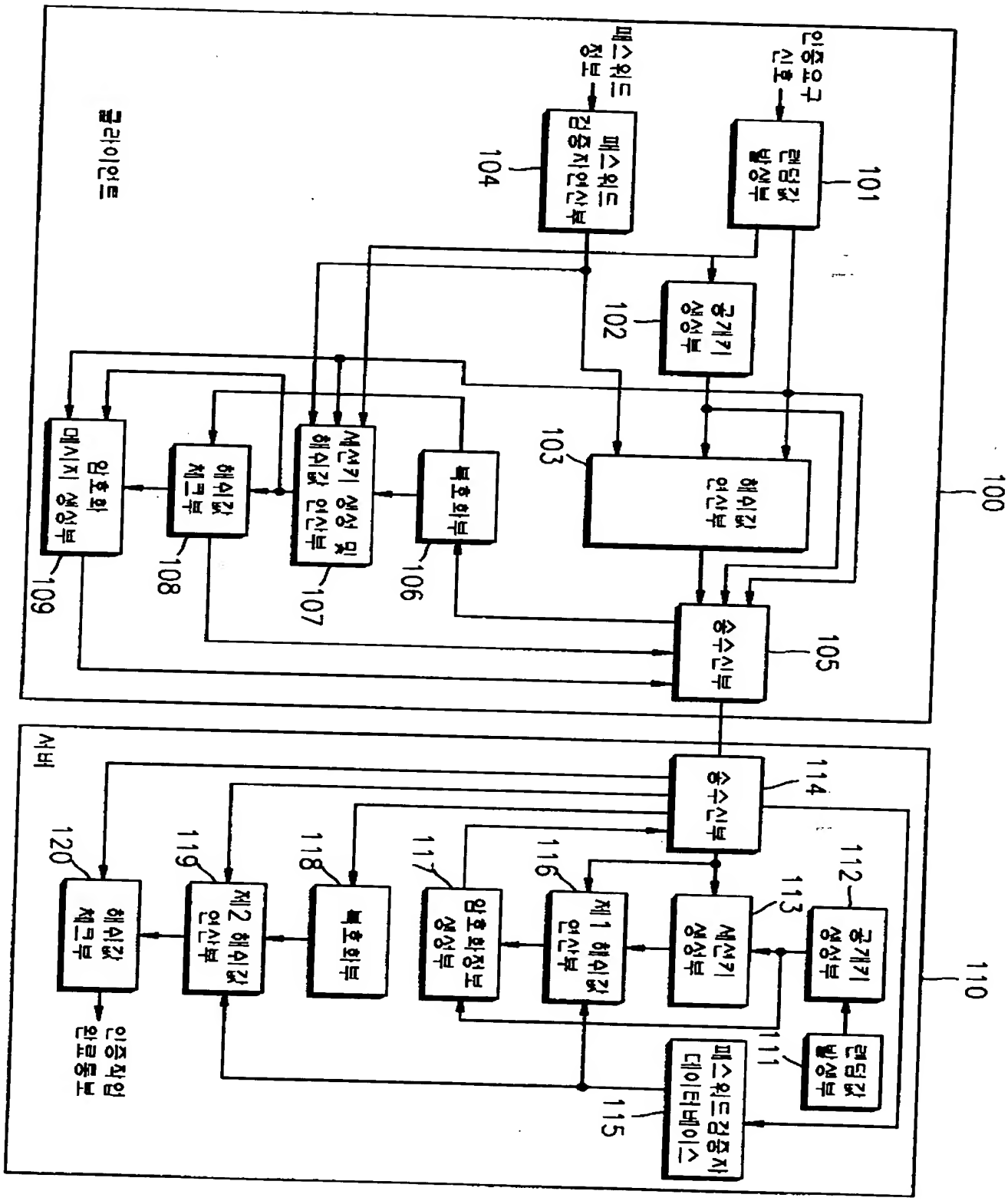
제6항에 있어서, 상기 클라이언트는, 사용자의 인증요구에 의해 상기 랜덤값을 발생하는 랜덤값 발생부, 상기 랜덤값 발생부에서 발생된 임시 개인키에 의해 공개키를 생성하는 공개키 생성부, 사용자로부터 패스워드 정보가 인가되면 대응되는 패스워드 검증자가 생성될 수 있도록 연산하는 패스워드 검증자 연산부, 상기 랜덤값 발생부에서 제공되는 랜덤값과 상기 공개키 생성부에서 제공되는 임시 공개키 및 상기 패스워드 검증자 연산부에서 제공되는 패스워드 검증자를 이용하여 상기 제 1 해쉬값을 얻기 위한 해쉬값 연산부, 상기 서버와 상호 인증 및 키교환을 위한 신호를 송수신하는 송수신부, 상기 송수신부를 통해 상기 서버로부터 수신된 상기 암호화 정보를 복호화하는 복호화부, 상기 복호화부로부터 제공되는 상기 서버에서 발생된 랜덤값과 상기 랜덤값 발생부에서 발생된 랜덤값을 이용하여 상기 세션키를 생성하고, 생성된 상기 세션키와 상기 패스워드 검증자 및 상기 랜덤값 발생부에서 발생되는 랜덤값을 이용하여 상기 제 2 해쉬값을 얻기 위한 세션키 생성 및 해쉬값 연산부, 상기 세션키 생성 및 해쉬값 연산부에서 제공되는 상기 제 2 해쉬값과 상기 복호화부에서 제공되는 복호화된 제 3 해쉬값을 비교하여 상기 서버를 인증하는 해쉬값 체크부, 상기 해쉬값 체크부로부터 상기 서버에 대한 인증작업 완료를 통보하는 메시지가 수신되면 상기 암호화 메시지를 생성하여 상기 송수신부로 전송하는 암호화 메시지 생성부를 포함하는 패스워드를 기반으로 한 상호 인증 및 키 교환장치.

청구항 8.

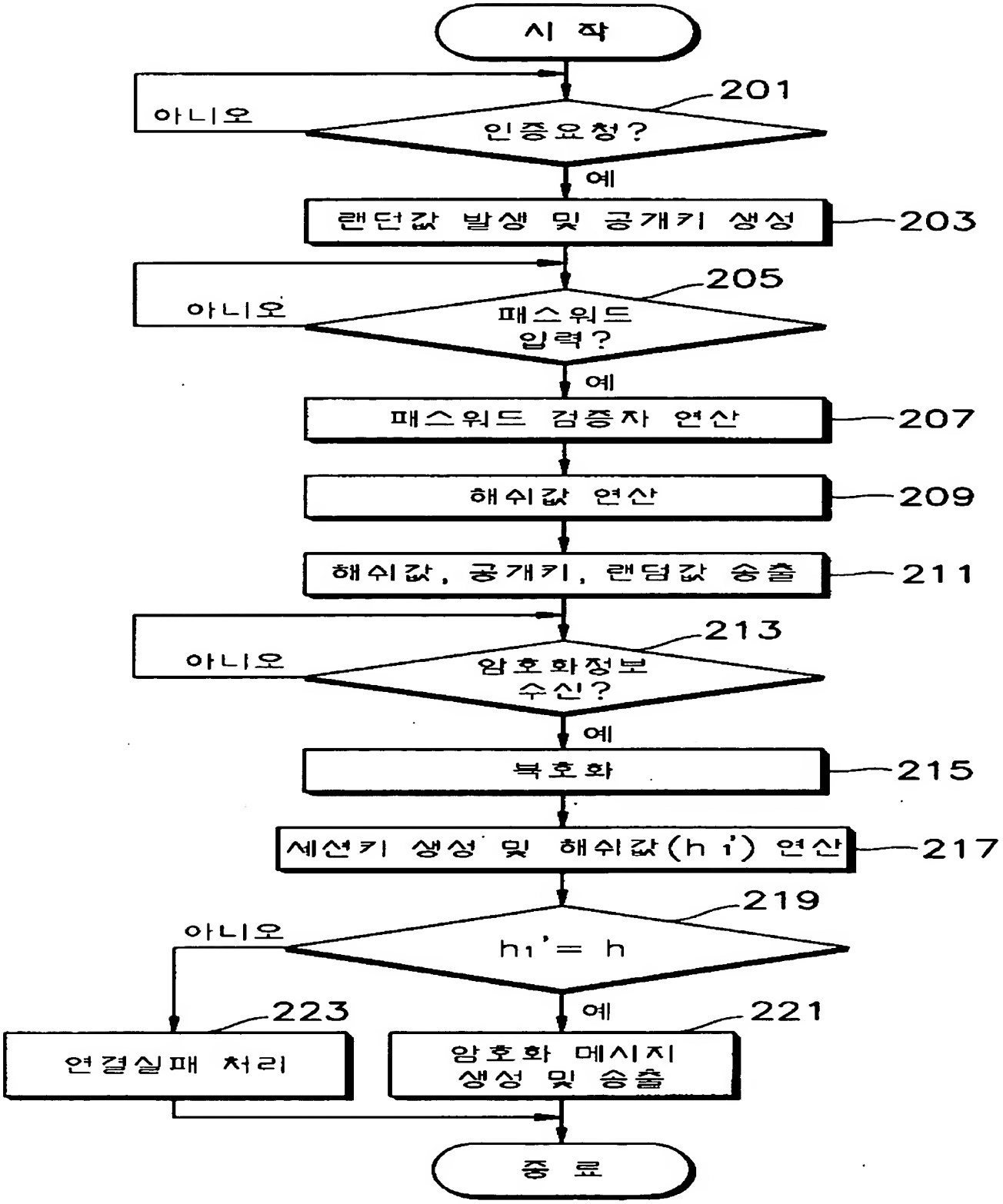
제6항 또는 제7항에 있어서, 상기 서버는, 상기 랜덤값을 발생하는 랜덤값 발생부, 상기 랜덤값 발생부에서 발생되는 임시 개인키에 의해 상기 서버의 임시 공개키를 생성하는 공개키 생성부, 상기 클라이언트와 상호 인증 및 키교환을 위한 신호를 송수신하기 위한 송수신부, 상기 송수신부를 통해 수신된 상기 클라이언트에서 발생된 랜덤값과 상기 공개키 생성부에서 발생된 임시 공개키를 이용하여 상기 클라이언트와의 세션키를 생성하는 세션키 생성부, 상기 송수신부를 통해 수신된 상기 클라이언트에서 발생된 랜덤값에 의해 대응되는 패스워드 검증자를 제공하는 패스워드 검증자 제공부, 상기 세션키 생성부에서 생성된 세션키와 상기 패스워드 검증자 제공부에서 제공되는 패스워드 검증자 및 상기 송수신부를 통해 수신된 상기 클라이언트에서 발생된 랜덤값을 이용하여 상기 제 3 해쉬값을 얻기 위한 연산을 수행하는 제 1 해쉬값 연산부, 상기 제 1 해쉬값 연산부에서 연산된 상기 제 3 해쉬값과 상기 서버의 임시 공개키를 암호화 정보로서 생성하여 상기 송수신부로 전송하는 암호화 정보 생성부, 상기 송수신부를 통해 수신된 상기 암호화 메시지를 복호화하는 복호화부, 상기 복호화부에서 복호화된 상기 클라이언트에서 발생된 랜덤값과 상기 패스워드 검증자 제공부에서 제공되는 패스워드 검증자를 이용하여 상기 제 4 해쉬값을 얻기 위한 연산을 수행하는 제 2 해쉬값 연산부, 상기 제 2 해쉬값 연산부에서 제공되는 상기 제 4 해쉬값과 상기 송수신부를 통해 수신된 상기 제 1 해쉬값을 비교하여 상기 클라이언트를 인증하는 해쉬값 체크부를 포함하는 패스워드를 기반으로 한 상호 인증 및 키 교환장치.

도면

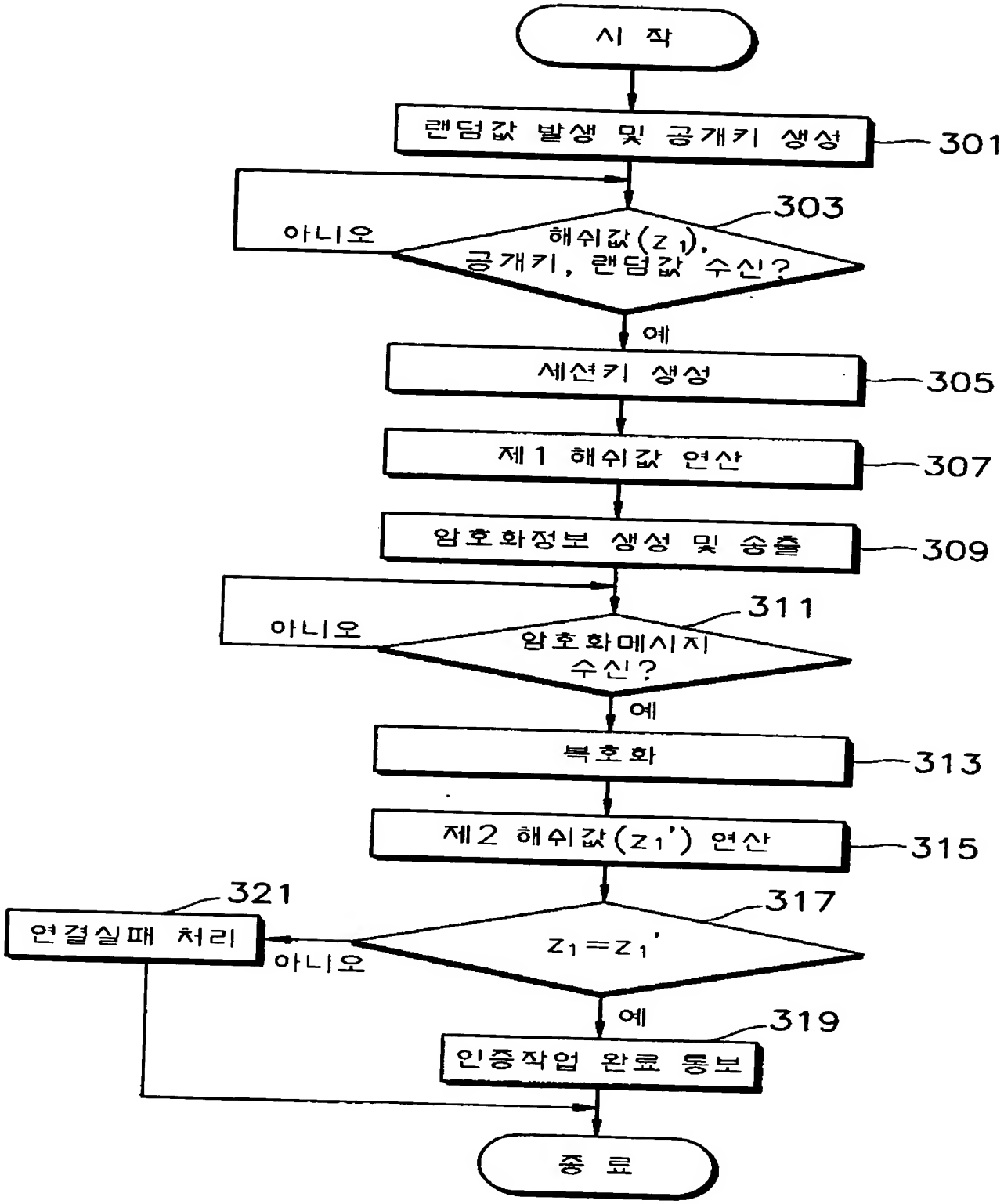
도면 1


클라이언트
패스워드 정보
인증요구 신호
패스워드 검증지연산부
104
클럭값 생성부
101
102
103
100
109
108
107
106
105
서버
114
112
113
116
117
118
119
120
111
115
110
인증작업 완료통보
클럭값 생성부
패스워드검증자 데이터베이스

도면 2


시 작
201
아니오
인증요청?
예
랜덤값 발생 및 공개키 생성
203
205
아니오
패스워드 입력?
예
패스워드 검증자 연산
207
해쉬값 연산
209
해쉬값, 공개키, 랜덤값 송출
211
213
아니오
암호화정보 수신?
예
복호화
215
세션키 생성 및 해쉬값(h i') 연산
217
219
아니오
h1'= h
예
223
221
연결실패 처리
암호화 메시지 생성 및 송출
종 료

도면 3


시 작
랜덤값 발생 및 공개키 생성
301
303
아니오
해쉬값(z1), 공개키, 랜덤값 수신?
예
세션키 생성
305
제1 해쉬값 연산
307
암호화정보 생성 및 송출
309
311
아니오
암호화메시지 수신?
예
복호화
313
제2 해쉬값(z1') 연산
315
321
317
연결실패 처리
아니오
z1=z1'
예
319
인증작업 완료 통보
종 료